



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **10016322 A**(43) Date of publication of application: **20.01.98**

(51) Int. Cl. **B41J 5/30**
G06F 3/12
H04N 1/393
H04N 1/41

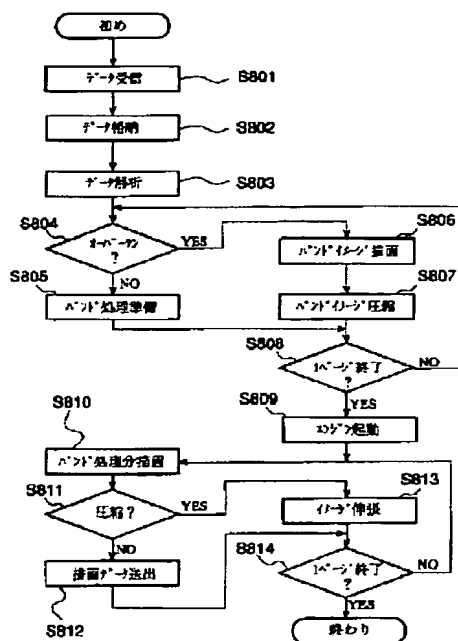
(21) Application number: **08176681**(71) Applicant: **CANON INC**(22) Date of filing: **05.07.96**(72) Inventor: **IKEDA JUN**(54) **IMAGE FORMATION METHOD AND APPARATUS THEREFOR**

(57) Abstract:

PROBLEM TO BE SOLVED: To provide an image formation method and an apparatus therefor whereby high-quality images can be realized in an inexpensive constitution without bringing about an overrun.

SOLUTION: Image data are converted to middle data whereby images can be formed (S801-S803). An image formation process time is estimated on the basis of the converted middle data (S804). When the estimated process time is longer than a predetermined time, the converted middle data are converted to bit map data (S804, S806). The obtained bit map data are compressed to generate compressed data (S807). When the estimated process time is shorter than the predetermined time, images are formed on the basis of the converted middle data. When it is not smaller, the generated compressed data are extended to form images (S809S814).

COPYRIGHT: (C)1998,JPO



(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 10-16322

(43) 公開日 平成10年(1998)1月20日

(51) Int. Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
B 4 1 J	5/30		B 4 1 J	5/30 Z
G 0 6 F	3/12		G 0 6 F	3/12 B
H 0 4 N	1/393		H 0 4 N	1/393
	1/41			1/41 Z

審査請求 未請求 請求項の数 13 O L

(全 16 頁)

(21) 出願番号 特願平 8-176681

(22) 出願日 平成8年(1996)7月5日

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 池田 純

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

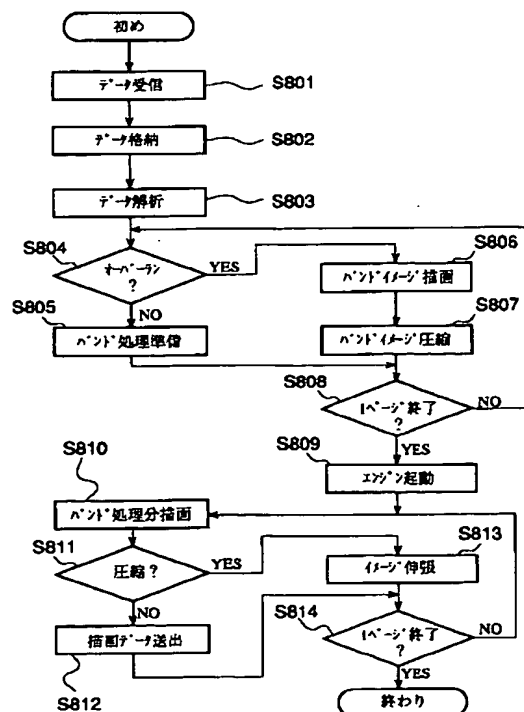
(74) 代理人 弁理士 大塚 康德 (外1名)

(54) 【発明の名称】 画像形成方法とその装置

(57) 【要約】

【課題】 オーバランが生ずることなく、高品位な画像形成を安価な構成で実現できる画像形成方法とその装置を提供する。

【解決手段】 画像データを、画像形成可能な中間データに変換する (S801-S803)。変換された中間データに基づいて画像形成処理時間を推定する (S804)。推定された画像形成処理時間が所定の時間より大きければ、変換された中間データをビットマップデータに変換する (S804, S806)。得られたビットマップデータを圧縮して圧縮データを生成する (S807)。推定された画像形成処理時間が所定の時間より小さければ、変換された中間データに基づいて画像形成を行い、そうでなければ、生成された圧縮データを伸張して画像形成を行う (S809-S814)。



【特許請求の範囲】

【請求項 1】 画像データを、画像形成可能な中間データに変換する第 1 変換工程と、
前記第 1 変換工程で変換された中間データに基づいて画像形成処理時間を推定する推定工程と、
前記推定工程で推定された画像形成処理時間が所定の時間より大きければ、前記第 1 変換工程で変換された中間データをビットマップデータに変換する第 2 変換工程と、
前記第 2 変換工程で得られたビットマップデータを圧縮して圧縮データを生成する圧縮工程と、
前記推定工程で推定された画像形成処理時間が所定の時間より小さければ、前記第 1 変換工程で変換された中間データに基づいて画像形成を行い、前記推定工程で推定された画像形成処理時間が所定の時間より大きければ、前記圧縮工程で生成された圧縮データを伸張して画像形成を行う画像形成工程とを備えることを特徴とする画像形成方法。

【請求項 2】 前記画像形成可能な中間データは、バンド単位の画像形成可能なデータであることを特徴とする請求項 1 に記載の画像形成方法。

【請求項 3】 前記推定工程は、前記第 1 変換工程で変換された中間データから対応するバンドイメージデータを生成するときの生成推定時間と、前記バンドイメージデータに基づいて画像形成するときの画像形成速度に基づいて、画像形成処理時間を推定することを特徴とする請求項 1 に記載の画像形成方法。

【請求項 4】 画像データを、画像形成可能な中間データに変換する第 1 変換工程と、
前記第 1 変換工程で変換された中間データの数が所定の数より大きければ、前記第 1 変換工程で変換された中間データをビットマップデータに変換する第 2 変換工程と、
前記第 2 変換工程で得られたビットマップデータを圧縮して圧縮データを生成する圧縮工程と、
前記第 1 変換工程で変換された中間データの数が所定の数より小さければ、前記第 1 変換工程で変換された中間データに基づいて画像形成を行い、前記第 1 変換工程で変換された中間データの数が所定の数より大きければ、前記圧縮工程で生成された圧縮データを伸張して画像形成を行う画像形成工程とを備えることを特徴とする画像形成方法。

【請求項 5】 前記画像形成可能な中間データは、バンド単位の画像形成可能なデータであることを特徴とする請求項 4 に記載の画像形成方法。

【請求項 6】 画像データを、画像形成可能な中間データに変換する第 1 変換手段と、
前記第 1 変換手段で変換された中間データに基づいて画像形成処理時間を推定する推定手段と、
前記推定手段で推定された画像形成処理時間が所定の時

間より大きければ、前記変換手段で変換された中間データをビットマップデータに変換する第 2 変換手段と、
前記第 2 変換手段で得られたビットマップデータを圧縮して圧縮データを生成する圧縮手段と、
前記推定手段で推定された画像形成処理時間が所定の時間より小さければ、前記第 1 変換手段で変換された中間データに基づいて画像形成を行い、前記推定手段で推定された画像形成処理時間が所定の時間より大きければ、前記圧縮手段で生成された圧縮データを伸張して画像形成を行う画像形成手段とを備えることを特徴とする画像形成装置。

【請求項 7】 前記画像形成可能な中間データは、バンド単位の画像形成可能なデータであることを特徴とする請求項 6 に記載の画像形成装置。

【請求項 8】 前記推定手段は、前記第 1 変換手段で変換された中間データから対応するバンドイメージデータを生成するときの生成推定時間と、前記バンドイメージデータに基づいて画像形成するときの画像形成速度に基づいて、画像形成処理時間を推定することを特徴とする請求項 6 に記載の画像形成装置。

【請求項 9】 画像データを、画像形成可能な中間データに変換する第 1 変換手段と、
前記第 1 変換手段で変換された中間データの数が所定の数より大きければ、前記第 1 変換手段で変換された中間データをビットマップデータに変換する第 2 変換手段と、
前記第 2 変換手段で得られたビットマップデータを圧縮して圧縮データを生成する圧縮手段と、
前記第 1 変換手段で変換された中間データの数が所定の数より小さければ、前記第 1 変換手段で変換された中間データに基づいて画像形成を行い、前記第 1 変換手段で変換された中間データの数が所定の数より大きければ、前記圧縮手段で生成された圧縮データを伸張して画像形成を行う画像形成手段とを備えることを特徴とする画像形成装置。

【請求項 10】 前記画像形成可能な中間データは、バンド単位の画像形成可能なデータであることを特徴とする請求項 9 に記載の画像形成装置。

【請求項 11】 請求項 6、または、請求項 9 のいずれか 1 つに記載の画像形成装置はページプリンタであることを特徴とする画像形成装置。

【請求項 12】 コンピュータプログラム製品であって、コンピュータ読み取り可能なプログラムコード手段を有するコンピュータ使用可能な媒体を備え、前記コンピュータプログラム製品は、
画像データを、画像形成可能な中間データに変換する、コンピュータ読み取り可能な第 1 プログラムコード手段と、
前記第 1 プログラムコード手段で変換された中間データに基づいて画像形成処理時間を推定する、コンピュータ

読み取り可能な第2プログラムコード手段と、
 前記第2プログラムコード手段で推定された画像形成処理時間が所定の時間より大きければ、前記第1プログラムコード手段で変換された中間データをビットマップデータに変換する、コンピュータ読み取り可能な第3プログラムコード手段と、
 前記第3プログラムコード手段で得られたビットマップデータを圧縮して圧縮データを生成する、コンピュータ読み取り可能な第4プログラムコード手段と、
 前記第2プログラムコード手段で推定された画像形成処理時間が所定の時間より小さければ、前記第1プログラムコード手段で変換された中間データに基づいて画像形成を行い、前記第2プログラムコード手段で推定された画像形成処理時間が所定の時間より大きければ、前記第4プログラムコード手段で生成された圧縮データを伸張して画像形成を行う、コンピュータ読み取り可能な第5プログラムコード手段とを備えることを特徴とするコンピュータプログラム製品。

【請求項13】 コンピュータプログラム製品であって、コンピュータ読み取り可能なプログラムコード手段を有するコンピュータ使用可能な媒体を備え、前記コンピュータプログラム製品は、
 画像データを、画像形成可能な中間データに変換する、コンピュータ読み取り可能な第1プログラムコード手段と、
 前記第1プログラムコード手段で変換された中間データの数が所定の数より大きければ、前記第1プログラムコード手段で変換された中間データをビットマップデータに変換する、コンピュータ読み取り可能な第2プログラムコード手段と、
 前記第2プログラムコード手段で得られたビットマップデータを圧縮して圧縮データを生成する、コンピュータ読み取り可能な第3プログラムコード手段と、
 前記第1プログラムコード手段で変換された中間データの数が所定の数より小さければ、前記第1プログラムコード手段で変換された中間データに基づいて画像形成を行い、前記第1プログラムコード手段で変換された中間データの数が所定の数より大きければ、前記第3プログラムコード手段で生成された圧縮データを伸張して画像形成を行う、コンピュータ読み取り可能な第4プログラムコード手段とを備えることを特徴とするコンピュータプログラム製品。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、画像形成方法とその装置、特に、ホストコンピュータなどのデータ供給源から描画元データを受信し、記録媒体に画像を形成する画像形成方法とその装置に関する。

【0002】

【従来の技術】 一般に、文書作成印字システム等の文書

処理、印字装置の概要は、図1の様に構成されており、図中、左から出力文書を組版等下降する処理装置であるホストコンピュータ101と、ホストコンピュータで加工された出力情報データを紙などの媒体に定着印字するプリンタ102を接続して構成している。

【0003】 具体的には、ユーザがホストコンピュータ101上にて、出力すべき画像を編集し、編集が終了された時点で、プリンタ102が受け付けられる出力画像用データに変換し、そのデータをプリンタ102へ送出する。プリンタ102は入力するデータに従い、紙面に印字する。図1中、プリンタコントローラが行なう処理例を、図2の構成図、及び、図3のフローチャートを用いて具体的に説明する。

【0004】 図2は、プリンタコントローラ内の様子を示した構成例の図である。201は、ホストコンピュータなどのデータ供給源からのデータを受信したり、ステータスのやりとりなどを行なうインターフェース部である。202は、プリンタ装置全体を制御／データ処理を行なう制御部（CPU）である。

【0005】 203は、プリンタコントローラ内に備わる主記憶装置であるDRAMをコントロールするメモリコントローラである。204は、主記憶であるメモリで、DRAMである一例を示している。205は、プリンタコントローラが印字時に描画データをプリンタエンジン207に送出するために備え付けられているDMAコントローラである。

【0006】 206は、DMA205経由で出力される描画イメージデータをプリンタエンジン207に送出するエンジンインターフェースである。207は、エンジンインターフェース206より出力された描画イメージデータを紙などの媒体に定着し、印字を実現するプリンタエンジンである。図3は、図2のプリンタコントローラの動作例を示したフローチャートである。まず、ステップS301では、ホストコンピュータなどのデータ供給源より描画元データを受信する。この描画元データは、具体的には、プリンタ機器に備わる固有のコードデータであったり、プリンタ機器に対するPDL等の描画言語であったりする。

【0007】 ステップS302では、受信されたデータをDRAM204内に格納する。一般に、インターフェースの通信速度とそのデータ処理との速度差を吸収するために、受信バッファと称されるDRAM204内に備わるメモリ領域に、一旦、バッファリングを行なう等するが、これがその動作に当たる。次に、ステップS303では、格納されたデータを解析し、イメージ描画の準備を行なう。この例では、バンド処理、具体的には、描画フルビットマップ領域のメモリ容量を持つことなく、紙面に対し描画バンドメモリ領域を持ち、そのバンドメモリデータに格納された描画イメージデータを印字しながら、その時間をもちいて、他のメモリエリアに割り当

てられた次のバンドイメージデータ格納領域に描画イメージデータを生成するダブルバッファ形式でバンド処理を実現している。そのために、バンド処理可能な中間描画データにデータ変換作業を行なう。

【0008】次に、ステップS304では、最初の印字バンドに相当するイメージバンドメモリに対し、イメージ描画を行なう。実際にはダブルバッファ形式であるため、最初だけは2つのバンドメモリに対しレンダリングを行なうことになる。レンダリングが終了したならば、ステップS305では、DMA205、並びにプリンタインターフェース206を印字可能状態にセットアップし、プリンタエンジン207に起動を掛ける。

【0009】そして、ステップS306では、すでに、ステップS304にてビットイメージ生成がバンドメモリに対して施されているバンド印字を行なう。一方、ステップS307では、プリンタエンジン207に、イメージデータをDMA205、エンジンインターフェース206を介して送出したDRAM204のバンドイメージ領域に対し、制御部202はメモリコントローラ203を経由して、次の印字位置に相当する描画データをレンダリングする。

【0010】ステップS308では、1バンドメモリ内の描画イメージデータの印字を終了したならば、1ページ分の印字が終了したかどうかを判断する。そして、1ページの印字を終了してなければ、ステップS309に進み、DMA205、エンジンインターフェース206を、次に印字するバンドメモリに切替え、次のバンド印字に備える。具体的には、DMA205のバンドイメージ転送終了割り込み等で、DMA転送アドレスの再設定、制御部202のラスタライズする描画バンドメモリアドレス等の変更を行なう。

【0011】バンド切り換えを行なったならば、DMA205、及び、エンジンインターフェース206は、次のバンドのイメージデータ転送をプリンタエンジン207に対して行ない、次のバンドの印字を実現すると共に、イメージデータ送出済のDRAM204内のバンドイメージ格納エリアに対し制御部202がラスタライズを行なうことを繰り返し、1ページ分の印字が実現される。

【0012】尚、描画バンドエリアとして、この場合2つのバンドメモリエリアを想定し、プリンタエンジン207に起動を掛ける前に、その2つのバンドメモリエリアに対し、レンダリングを終了（ステップS304）する説明をしたが、正確な動作シーケンスとしては、始めのバンド印字に際しては、イメージデータを送出しつつラスタライズするという並行動作は起こらない。また、1ページ終了前の動作シーケンスについても、並行動作が起こらないことになるのは言うまでもない。

【0013】また、並行動作時のDRAM205からのデータ要求と、制御部202からのラスタライズの各D

RAMへ対するアクセスアービトレーションは、メモリコントローラ203が行なっている。このようなバンド処理を行なうことで、1ページ分のビットマップメモリを持たずに、1ページ分の印字を実現可能とする機器に於いては、レンダリングする画像が複雑になったり、プリンタエンジンの印字速度が高速になってくる等の各要因により、並行動作中のラスタライズ処理が印字データ送出に時間的に間に合わなくなる事態（以下、この事態をオーバーランと記す）が生ずる。

10 【0014】これに対応するためには、ページ分のイメージデータ格納の為のメモリエリア、いわゆる、ビットマップ（1ピクセル8ビット多値のメモリであれば、バイトマップ）メモリを保持することで、並行動作を回避する処理系も存在している（いわゆるフルページメモリを搭載した処理系）。また、更にメモリ効率を上げ、コストを抑えるために圧縮技術を導入して対処している機器も登場している。

20 【0015】図4が、その様な圧縮／伸長技術を導入して機器を構築した場合の構成例の図である。401は、ホストコンピュータなどのデータ供給源からのデータを受信したり、ステータスのやりとりなどを行なうインターフェース部である。402は、プリンタ装置全体を制御／データ処理を行なう制御部である。

30 【0016】403は、プリンタコントローラ内の主記憶装置であるDRAMをコントロールするメモリコントローラである。メモリコントローラ403は、圧縮伸長部405と制御部402間のDRAM404に対するアクセス競合のアービトレーションを行う。404は、主記憶であるメモリで、DRAMである場合を示している。

【0017】405は、イメージデータの圧縮／伸長を行なう部分である。406は、プリンタコントローラが印字時に描画データを、圧縮／伸長部405にて伸長されたデータを、プリンタエンジン408に送出するため

40 7より出力された描画イメージデータを紙などの媒体に定着し、印字を実現するプリンタエンジンである。次に、図5は、図4に示した圧縮／伸長技術を用いた処理動作を示すフローチャートであり、これを用いてその流れを示す。まず、ホストコンピュータなどのデータ供給源より描画元データ（具体的にはプリンタ機器に備わる固有のコードデータであったり、昨今はPDLと称され、プリンタ機器に対する描画言語であったりする）を受信し（ステップS501）、受信されたデータをDRAM404内に格納する（ステップS502）。一般にインターフェースの通信速度とそのデータ処理との速度

差を吸収するために、受信バッファと称されるDRAM 404内に備わるメモリ領域に一旦バッファリングを行なう等するが、これがその動作に当たる。

【0019】次に格納されたデータを解析しイメージ描画の準備を行なう(ステップS503)。前述の圧縮／伸長技術を用いない従来例では、バンド処理、具体的には描画フルビットマップ領域のメモリ容量を持つことなく、紙面に対し描画バンドメモリ領域を持ち、そのバンドメモリデータに格納された描画イメージデータを印字しながら、その時間を持ちいて、他のメモリエリアに割り当てられた次のバンドイメージデータ格納領域に描画イメージデータを生成、ダブルバッファ形式でバンド処理を実現しており、そのためにバンド処理可能な中間描画データにデータの変換作業を行っていたが、圧縮／伸長技術を用いてメモリ削減を実現する場合についても、メモリ削減のために1ページ分の描画メモリエリアを備えずに印字可能とするわけであるから、当然全てのデータを描画終了後、圧縮を開始することは出来ず、1ページ分の描画に際しては、印字領域をバンド処理と同様に複数のエリアに分け、そのエリアをレンダリングし、圧縮、格納という動作を繰り返し、1ページ分の圧縮イメージデータの生成を行なう必要があり、何らかのバンド処理同等のデータ変換作業を行なうことになる。

【0020】次に最初の印字バンドに相当するイメージバンドメモリに対し、イメージ描画を行なう(ステップS504)。実際にはDRAM404内に備わるバンドメモリ領域に対して、この印字位置に相当するラスター化を行なうことになる。尚、前述図2に示したバンド並行処理のようにダブルバッファにする必要は特に無い。

【0021】1バンドのレンダリングが終了したならば、そのレンダリングされたイメージデータを圧縮し、やはりDRAM404内に備わる圧縮データ格納エリアに格納する(ステップS505)。ステップS504、ステップS505の動作を1ページ分の印字データの生成／圧縮／格納が終了するまで行ない、1ページ分の圧縮イメージデータの生成／格納を終了する(ステップS506)。

【0022】1ページ分の圧縮イメージデータが生成されたならば、圧縮／伸長部405、DMA406、並びにプリンタインターフェース407を印字可能状態にセットアップし、プリンタエンジン408に起動を掛ける(ステップS507)。起動が掛かったエンジンは同期信号を発生し、その同期信号に従いエンジンインターフェース407を介して、DMA406は圧縮／伸長部405にデータを要求、一方圧縮／伸長部405は、DRAM404内に格納されている1ページ分の圧縮イメージデータをメモリコントローラ403を介して読み込み、圧縮データを伸長し、印字すべきイメージデータを復元する(ステップS508)。

【0023】508の処理を1ページ分の印字が終了するまで繰り返し(ステップS509)、1ページ分の印字出力が実現される。

【0024】

【発明が解決しようとする課題】しかしながら、上述の従来例では、以下に示す問題がある。まず、前述のバンド処理では、描画する内容の複雑度とプリンタエンジン部の印字速度のかねあいから、オーバーランが発生する可能性があり、この問題に対処するためには、メモリを増設したり、解像度を落とすなどの方策を取らざるを得ないため、追加メモリ分のコストが上昇したり、画質が劣化してしまうなどの問題があった。

【0025】一方、前述の圧縮／伸長処理については、前述のようにプリンタエンジンの動作速度での伸長処理速度だけを満足すれば、基本的にオーバーランの事態は避けられるものの、スループットを向上させるためには、圧縮／伸長部の同時動作が可能でなければならない。そのため、2つの圧縮／伸長部を備えることが必要となり、装置構成が高価なものになってしまう。即ち、従来の画像形成装置では、コストダウンとスループットの向上のバランスをとることが困難であった。

【0026】本発明は、上記従来例に鑑みてなされたもので、オーバーランが生ずることなく、高品位な画像形成を安価な構成で実現できる画像形成方法とその装置を提供することを目的とする。

【0027】

【課題を解決するための手段】上記目的を達成するため、本発明の画像形成方法とその装置は以下の構成を備える。即ち、画像データを、画像形成可能な中間データに変換する第1変換工程と、前記第1変換工程で変換された中間データに基づいて画像形成処理時間を推定する推定工程と、前記推定工程で推定された画像形成処理時間が所定の時間より大きければ、前記第1変換工程で変換された中間データをビットマップデータに変換する第2変換工程と、前記第2変換工程で得られたビットマップデータを圧縮して圧縮データを生成する圧縮工程と、前記推定工程で推定された画像形成処理時間が所定の時間より小さければ、前記第1変換工程で変換された中間データに基づいて画像形成を行い、前記推定工程で推定された画像形成処理時間が所定の時間より大きければ、前記圧縮工程で生成された圧縮データを伸張して画像形成を行う画像形成工程とを備える。

【0028】また、別の発明は、画像データを、画像形成可能な中間データに変換する第1変換工程と、前記第1変換工程で変換された中間データの数が所定の数より大きければ、前記第1変換工程で変換された中間データをビットマップデータに変換する第2変換工程と、前記第2変換工程で得られたビットマップデータを圧縮して圧縮データを生成する圧縮工程と、前記第1変換工程で変換された中間データの数が所定の数より小さければ、

前記第1変換工程で変換された中間データに基づいて画像形成を行い、前記第1変換工程で変換された中間データの数が所定の数より大きければ、前記圧縮工程で生成された圧縮データを伸張して画像形成を行う画像形成工程とを備える。

【0029】また、別の発明は、画像データを、画像形成可能な中間データに変換する第1変換手段と、前記第1変換手段で変換された中間データに基づいて画像形成処理時間を推定する推定手段と、前記推定手段で推定された画像形成処理時間が所定の時間より大きければ、前記第1変換手段で変換された中間データをビットマップデータに変換する第2変換手段と、前記第2変換手段で得られたビットマップデータを圧縮して圧縮データを生成する圧縮手段と、前記推定手段で推定された画像形成処理時間が所定の時間より小さければ、前記第1変換手段で変換された中間データに基づいて画像形成を行い、前記推定手段で推定された画像形成処理時間が所定の時間より大きければ、前記圧縮手段で生成された圧縮データを伸張して画像形成を行う画像形成手段とを備える。

【0030】また、別の発明は、画像データを、画像形成可能な中間データに変換する第1変換手段と、前記第1変換手段で変換された中間データの数が所定の数より大きければ、前記第1変換手段で変換された中間データをビットマップデータに変換する第2変換手段と、前記第2変換手段で得られたビットマップデータを圧縮して圧縮データを生成する圧縮手段と、前記第1変換手段で変換された中間データの数が所定の数より小さければ、前記第1変換手段で変換された中間データに基づいて画像形成を行い、前記第1変換手段で変換された中間データの数が所定の数より大きければ、前記圧縮手段で生成された圧縮データを伸張して画像形成を行う画像形成手段とを備える。

【0031】また、別の発明は、コンピュータプログラム製品であって、コンピュータ読み取り可能なプログラムコード手段を有するコンピュータ使用可能な媒体を備え、前記コンピュータプログラム製品は、画像データを、画像形成可能な中間データに変換する、コンピュータ読み取り可能な第1プログラムコード手段と、前記第1プログラムコード手段で変換された中間データに基づいて画像形成処理時間を推定する、コンピュータ読み取り可能な第2プログラムコード手段と、前記第2プログラムコード手段で推定された画像形成処理時間が所定の時間より大きければ、前記第1プログラムコード手段で変換された中間データをビットマップデータに変換する、コンピュータ読み取り可能な第3プログラムコード手段と、前記第3プログラムコード手段で得られたビットマップデータを圧縮して圧縮データを生成する、コンピュータ読み取り可能な第4プログラムコード手段と、前記第2プログラムコード手段で推定された画像形成処理時間が所定の時間より小さければ、前記第1プログラ

ムコード手段で変換された中間データに基づいて画像形成を行い、前記第2プログラムコード手段で推定された画像形成処理時間が所定の時間より大きければ、前記第4プログラムコード手段で生成された圧縮データを伸張して画像形成を行う、コンピュータ読み取り可能な第5プログラムコード手段とを備える。

【0032】また、別の発明は、コンピュータプログラム製品であって、コンピュータ読み取り可能なプログラムコード手段を有するコンピュータ使用可能な媒体を備え、前記コンピュータプログラム製品は、画像データを、画像形成可能な中間データに変換する、コンピュータ読み取り可能な第1プログラムコード手段と、前記第1プログラムコード手段で変換された中間データの数が所定の数より大きければ、前記第1プログラムコード手段で変換された中間データをビットマップデータに変換する、コンピュータ読み取り可能な第2プログラムコード手段と、前記第2プログラムコード手段で得られたビットマップデータを圧縮して圧縮データを生成する、コンピュータ読み取り可能な第3プログラムコード手段と、前記第1プログラムコード手段で変換された中間データの数が所定の数より小さければ、前記第1プログラムコード手段で変換された中間データに基づいて画像形成を行い、前記第1プログラムコード手段で変換された中間データの数が所定の数より大きければ、前記第3プログラムコード手段で生成された圧縮データを伸張して画像形成を行う、コンピュータ読み取り可能な第4プログラムコード手段とを備える。

【0033】

【発明の実施の形態】はじめに、本発明の実施の形態の画像形成方法とその装置のポイントを要約した後に、その詳細な説明に入るものとする。本発明の実施の形態の画像形成方法とその装置は、描画されるデータ内容や、ユーザーの機器に対する動作指定、機器に備わるメモリ容量（ユーザーの拡張などで変化する）、インターフェース接続状況（接続インターフェースの速度や、複数の接続など）等の各要素に対応して、最適な描画印字処理方法を選択して実行する。

【0034】以上のような構成により、低コストでオーバーランの生じない、かつ、できる限りスループットの低下を招かないプリンタを構築することが可能である。以下、本発明の実施の形態の画像形成方法とその装置の詳細な説明を行う。図6は、本発明の一実施の形態であるプリンタコントローラの概要を示した構成図であり、以下の各部から構成されている。

【0035】601は、ホストコンピュータなどのデータ供給源からのデータを受信したり、ステータスのやりとりなどを行なうインターフェース部である。具体的には、パラレルインターフェース、シリアルインターフェース、ネットワークインターフェース等である。602は、プリンタ装置全体の制御／データ処理を行なう制御

部（CPU）であり、入力描画命令データの解析、描画イメージデータの生成、並びに、機器に備わる各部を制御する。

【0036】603は、主記憶装置であるDRAM604をコントロールするメモリコントローラである。メモリコントローラ603は、セクタ部606からのデータアクセスに対応して、DRAM604をアクセスする。また、DRAM604のリフレッシュ処理等を行う。尚、DRAM604は、ダイナミックRAMに限定されることはなく、制御部602で実行される制御プログラムを格納するメモリであれば、他の種類のメモリであってもよいことは言うまでもない。

【0037】DRAM604は、ホストからの描画命令データに従い、制御部602によって生成された描画イメージデータを格納したり、圧縮／伸長部605で圧縮されたデータを格納したり、作業メモリとして使用される。605は、イメージデータの圧縮／伸長を行なう処理部である。ここでの圧縮／伸長アルゴリズム自体は特に限定するものではないが、イメージデータを、メモリ削減可能な圧縮率で圧縮可能な圧縮アルゴリズムと、圧縮されたデータを伸張する場合は、エンジン609のデータ転送スピードに間にあうように伸張する伸張アルゴリズムを備えている。尚、具体的圧縮アルゴリズムの例としては、JPEG、JBIG等が挙げられる。

【0038】606は、データバスを切り換えるセクタ部であり、バンド処理、圧縮処理、伸長印字処理の3つの動作に対応してデータバスを切り換える。そのデータバスの切り換え制御は、制御部602が行う。607はDMAコントローラであり、プリンタコントローラが印字時に、セクタ部606で選択されたデータバスのデータをプリンタエンジン609に送出するためのDMA制御を行う。DMAコントローラ607の処理動作は、制御部602の設定に従う。

【0039】608は、DMA607経由で出力される描画イメージデータをプリンタエンジン609に送出するためのエンジンインターフェースである。エンジンインターフェース608の処理動作は、制御部602によって制御される。609は、エンジンインターフェース608より出力された描画イメージデータを紙などの媒体に定着し、印字を実現するプリンタエンジンである。プリンタエンジン609は、例えば、LBPに代表されるページプリンタエンジンである。

【0040】次に、図7は、セクタ606、及び、その周辺の接続状況、動作状況を説明するための図である。701は、制御部602（図6）に相当する制御部（CPU）である。702は、図6の圧縮／伸長部606に相当する圧縮／伸長部である。703は、図6のセクタ606に相当するセクタ部である。セクタ部703は、制御部701からの指示に基づいて、L1からL7の各バスのいずれかのバスを確立する。

【0041】704は、図6のメモリコントローラ603に相当するメモリコントローラ部である。705は、図6のDMA607に相当するDMA部である。次に、図8は、本発明の1実施の形態の制御の流れを示すフローチャートで、本発明の特徴に係るバンド処理と圧縮／伸長処理との切り換え制御に関する。ここでは、オーバーラン予測されたものについては、圧縮／伸長処理を行う。また、オーバーラン予測されなかったものについては、バンド処理を行なう。以下、この制御手順について説明する。

【0042】まず、ステップS801では、ホストコンピュータなどのデータ供給源より描画データを受信する。この描画データは、例えば、プリンタ機器に備わる固有のコードデータであったり、プリンタ機器に対する描画言語（PDL）であったりする。ステップS802では、制御部602は、受信されたデータをセクタ606、メモリコントローラ603経由にてDRAM604内に格納する。

【0043】これは、インターフェースの通信速度とそのデータ処理との速度差を吸収するために、DRAM604内に備わるメモリ領域（受信バッファ）にバッファリングされるものである。ステップS803では、制御部602は、DRAM604内のバッファに格納された描画データを解析して、イメージ描画の準備を行なう。ここでは、以下のステップで実行されるバンド処理を可能とするために、描画データを中間描画データへ変換する。

【0044】本実施の形態の、描画フルビットマップ領域分のメモリ容量を持つことなく実行可能なバンド処理構成をここで、簡単に要約する。まず、印字紙面に対する描画バンドメモリ領域を持ち、その領域に格納された描画イメージデータを読み出して、印字する。また一方では、その印字中に、他のメモリエリアに割り当てられた次のバンドイメージデータの格納領域に描画イメージデータを生成するという、いわゆる、ダブルバッファ形式でのバンド処理を行う。

【0045】また、圧縮／伸長をフルメモリ（ビットマップ／バイトマップ等）を必要とすることなく行なうために、描画処理単位をバンド単位で行なっており、そのために、描画データをバンド処理可能な中間描画データに変換する必要がある。尚、上述のバンドメモリの容量は、機器に備わるメモリ容量、プリンタエンジンのデータレート等から決定される。

【0046】次に、ステップS804では、ステップS803で生成されたバンド毎の中間コードをスキャンし、そのバンドイメージデータ生成に制御部602が費やすであろう処理時間を計算する。そして、この処理時間に基づいて、プリンタエンジンへの転送速度を鑑み、オーバーランが生じるか否かを判定する。そして、オーバーランが生じると予測した場合は、ステップS807へ進

み、そうでない場合は、ステップS805へと進む。

【0047】尚、ここでのオーバーラン予測法については特に限定しないが、例えば、以下の方法で実現可能である。即ち、バンド描画のための中間コードを解析し、その中間コードの処理単位に掛かる描画時間とその構成要素の総合計を算出する。そして、その結果から、バンドイメージ描画に掛かる描画時間を算出し、この値と前もってわかっている本プリンタコントローラに接続されるプリンタエンジンのデータ転送レートとからオーバーランの発生を予測する。

【0048】言うまでもないが、プリンタコントローラ内のバス構成や、コントローラ内の他のバスマスタなどの動作状況により、バスの競合に由来する処理時間の変化が生じるが、これらは盛り込まれているものとし、ここでは説明を割愛する。次に、ステップS805では、後述するステップS809にてエンジンに起動を掛けた場合での処理シーケンスを記憶しておくメモリキューに対して、そのバンドの中間コードデータを登録する。ここで、処理シーケンスとは、 SHIPPING に際して行う処理であり、

- 1 中間コード、データ長、ソースアドレス、ディスティネーションアドレス等をラスタライザに設定
- 2 圧縮データ格納アドレス、ディスティネーションアドレス、データ長等を伸張手段に設定する等の処理シーケンスである。この処理シーケンスは、バンド毎にキューに対して登録され、ステップS809にて、エンジンに起動を掛けた後、キューの内容に従いバンド毎にプリントしつつ、イメージデータの生成を行う。

【0049】ここで、登録内容については特に限定するものではないが、例えば、バンドの中間コードデータのデータ長、記憶先頭アドレス等を登録し、そのバンドのデータ転送のタイミングに合わせて指定された中間コードデータを処理する形になる。ステップS808では、バンド処理の登録が1ページ分終了したか否かを判断し、まだであるならば、次に描画するバンドについて処理するために、ステップS804へ進む。また、終了ならば、印字動作を開始するために、ステップS809へ進む。

【0050】一方、ステップS804にてオーバーランと予測された場合は、ステップS806に進み、制御部602が、そのバンドについてのイメージ描画を行なう。ここでの描画動作処理は、バンド処理時のラスタライズと同等なものである。具体的には、セクタ606、メモリコントローラ603を介して、DRAM604内に格納されている中間コードを制御部602が読み込み、そのコード内容に対応してイメージデータを生成する。

【0051】ステップS807では、セクタ606を切り替えて、制御部602から圧縮／伸長部605に至るバス（圧縮処理バス）を確立する。そして、この確立

されたバスを通じて、生成されたイメージデータを圧縮／伸長部605に送り、圧縮／伸長部605では、このイメージデータの圧縮を行う。その後、また、セクタ606を介して、圧縮されたイメージデータをDRAM604内に備わる圧縮バンドイメージ格納エリアに格納する。

【0052】ここで、具体的なデータの流れを図7を参照して説明する。制御部602からDRAM604へアクセスする場合は、セクタ部606は、L4、L5、L6の各バスを確立する。また、圧縮処理時には、制御部602は、抽出／伸長部605（702）を圧縮モードに設定すると共に、セクタ部606に対して、L1、L2の各バスを確立するように要求し、セクタ部606は、L1、L2の各バスを確立する。

【0053】このように、1つのバンドエリアの圧縮処理は、ステップS806／ステップS807の処理ステップを、上述のバスの制御に同期して繰り返すことで実現される。尚、この圧縮処理の実現については本実施の形態で示す処理順序に限定されるわけではなく、例えば、一旦、描画バンドイメージをDRAM604内に格納してバンドイメージ生成が終了してから、バンドイメージ全体について一括して圧縮処理を施してもよい。その場合のセクタ606、即ち、セクタ部703内のデータバスは図9に示す様になる。

【0054】即ち、非圧縮／伸張処理時には、G1、G2、G6の各バスが確立される。また、圧縮処理時には、G4、G5の各バスが確立される。さらに、伸張処理時には、G3、G7の各バスが確立される。セクタ606、セクタ部703、セクタ部903の切り換え設定方法については、特に特定の方法に限定されるわけではなく、例えば、圧縮格納バスをメモリマップに割り当て、バスのアービトレーションを適宜に施すことで実現可能である。

【0055】次に、図8のフローチャートの各処理ステップでの説明に戻る。ステップS808では、ステップS805のステップからの流れと同様に1ページ分のバンドイメージについての処理が終了したか否かを判断する。そして、終了ならば、ステップS809へ進む。また、そうでない場合は、ステップS804へ戻る。

【0056】以上説明したステップS804～ステップS808の各ステップを繰り返した後の状態は以下のようになる。即ち、1ページ分の描画に必要なデータが上述の中間コードとしてキューイングされたものと、圧縮されたバンドイメージデータがその描画順序に従ってキューイングされた状態にある。尚、上述の中間コードは、制御部602が高速にバンドイメージを描画（レンダリング）可能なコードである。

【0057】このように、バンドイメージデータ発生に必要なデータが完成した状態になると、ステップS809では、次の印字をするために、プリンタエンジンを起

動する。次に、ステップS810では、プリンタエンジンに起動が掛かると、制御部602は、印字処理を開始する。即ち、制御部602は、上述したキューイングされている中間コードに基づいて、DRAM604のバンドバッファに対して、イメージ描画（レンダリング）を開始する。

【0058】この場合、セクタ606、メモリコントローラ603経由のアクセスとなり、セクタ606はL4、L5、L6の各データパスを確立する。尚、プリンタエンジンのデータレートにも依存するが、圧縮されたバンドイメージについては、その圧縮データポイント等がキューイングされていて、非圧縮のバンド描画中間コードについては、ダブルバッファ構成にて処理を行なう場合、2バンド分のイメージ描画（レンダリング）を先行し、そのレンダリングが終了してからプリンタエンジンに起動をかける方が、オーバランに対しては有利である。

【0059】次に、ステップS811では、現状の印字位置に対する描画データソースをキューイングされたデータ種別をチェックし、圧縮されているデータソースの印字であれば、ステップS813へ進む。そうでなければ、ステップS812へと進む。ステップS812では、そのバンド印字については、ステップS810で生成されているイメージデータ、もしくは、ステップS809でのエンジンへの起動前に生成されているバンドイメージデータを、エンジン609側に送出して、1バンドイメージの印字を行う。尚、この送出は、セクタ606、DMA607、エンジンインターフェース608を介して行う。この場合でのセクタ606で選択されるデータパスは、非圧縮／伸長パス（図7）となる。次に、ステップS814へ進む。

【0060】一方、ステップS811の判定で圧縮データであった場合での処理ステップS813では、DRAM604の圧縮データ格納エリアに格納されたデータをソースデータとし、セクタ606を伸長処理時のパス（図7）に設定する。そして、DMA607（705）、エンジンインターフェース608を用いて、エンジン609のデータ転送レートに合わせて、リアルタイムで伸長処理を施しつつ、1バンドイメージ印字を実現する。

【0061】ステップS812、もしくは、ステップステップS813のいずれかによって1バンドイメージの印字を終了したならば、ステップS814に進む。ステップS814では、1ページ分のバンドイメージの印字が終了したか否かを判断し、終了してなければ、ステップS810へ戻り、1ページ終了であれば1ページ分の印字処理の終了となる。

【0062】ここで、ステップS810に戻ると、その時点で印字終了したバンドイメージ印字処理が伸長処理であった場合（ステップS813）は、ステップS

810はスルーする。また、非伸長処理（ステップS812）であった場合は、既に前のステップS812にてバンドイメージの印字を終了しているエリアに対してキューイングされた処理が中間コードからのラスタライズを含む場合、次のバンドイメージ生成処理（ラスタライズ処理）を、制御部602がバックグラウンドにて行なう。

【0063】一方、バックグラウンドによるラスタライズ処理の最中には、次のバンド印字のためにダブルバッファに既に生成されているイメージデータをステップS812を通じて印字するか、ステップS813のステップでリアルタイムの伸長処理印字を行なっているかのいずれかの状態になる。従って、ステップS812のステップを通じて印字処理を行なうか、ステップS813のステップを通じて印字処理を行なうかについては、印字を行なうバンドイメージの描画の質に依存し、概して、複雑な図形を印字するのであれば、ステップS813のステップが多く、そうでない場合は、ステップS812のステップが多くなる。

【0064】いずれにせよ、全ての印字プロセスがステップS813を通るケースはよほど複雑な印字データが連続する場合に限られるわけであり、これらステップS812の処理とステップS813の処理を混在して処理できるような本実施の形態の実現により、1ページ分の圧縮データ格納エリアの確保によるメモリフルとなり、結局、次のページの処理の開始ができなくなる事態を積極的に回避することができる。また、メモリ効率を上げ、かつ、オーバランの危険性も回避できる機器構成が可能となり、さらに、スループットの向上も期待できる。

【0065】尚、本実施の形態でのイメージ描画処理（ラスタライズ）については制御部602を用いて実現していたが、ハードウェアを用いたラスタライズプロセッサ等を用いても同様であり（この場合、処理速度向上によって、ステップS812の処理ステップ頻度が上がる）、特に限定されるものではない。

【他の実施の形態】 上述の実施の形態では、圧縮／伸長処理と非圧縮／伸長処理の切り換えについて、制御部602によるイメージ描画（ラスタライズ）時間を描画するバンドについて調べ、その結果に従って処理を切り換えていた。しかし、これを、例えば、印字されるデータの複雑度を数量化し、それに基づいて処理を切り換えることも可能である。

【0066】以下、この処理を実現する処理手順を図10のフローチャートを参照して説明する。まず、ホストコンピュータなどのデータ供給源より描画元データを受信し（ステップS1001）、制御部602は、受信されたデータをセクタ606、メモリコントローラ603経由にて、DRAM604内に格納する（ステップS1002）。

【0067】次に、制御部602は、DRAM604内のバッファに格納された描画元データを解析し、イメージ描画の準備を行なう（ステップS1003）。次に、ステップS1003のステップでバンド処理の為の描画中間コードのソートが施されたところで、その中間コード数を勘定し、あらかじめ決められ保持されているオーバラン発生の限界中間コード数との比較を行なう（ステップS1004）。

【0068】実際には、制御部602が印字時にできるだけ高速にイメージ生成（ラスターライズ）を行えるように、中間コード自体は単純な作業の集合命令になることが予想され、コード数そのものから処理時間を近似できる場合が多い。従って、中間コード命令数そのものでオーバランをある程度予測できる場合であれば、制御部602による描画イメージデータ生成（ラスターライズ）に掛かる時間を正確に算出するよりも、処理を高速化できる。

【0069】尚、処理速度とのトレードオフを考慮の上で、中間コードの命令の種類、データ量を加味して、更に近似値を正確なものにすることは容易に可能である。次に、ステップS1005では、ステップS1004にて比較した内容が、所定の設定値よりも多くの中間コードを含かどうかチェックする。すなわち、この方法でオーバランが生じると予測された場合は、ステップS1007へ進む。逆に、そうでない場合は、ステップS1006へ進む。

【0070】ステップS1006では、後述するステップS1010にてエンジンに起動を掛けた場合の処理シーケンスを記憶しておくキューに対し、そのバンドの中間コードデータを登録する。次に、ステップS1009に進む。ステップS1009では、バンド処理の登録が1ページ分終了したか否かを判断する。そして、まだであるならば、次に描画するバンドについて処理するために、ステップS1004へ、逆に終了ならば、印字動作を開始するためにステップS1010へと進む。

【0071】一方、ステップS1005にてオーバランと予測された場合は、ステップS1007に進み、制御部602が、そのバンドについてイメージ描画を行なう。ステップS1008では、ステップS1007で精製されたイメージを、圧縮／伸長部605に送り、圧縮／伸長部605では、このイメージデータの圧縮を行う。そして、圧縮されたイメージデータをDRAM604内に備わる圧縮バンドイメージ格納エリアに格納する。

【0072】ステップS1009では、1ページ分のバンドイメージについての処理が終了したか否かを判断する。そして、終了ならば、ステップS1010へ進み、そうでない場合はステップS1004へ戻る。以上説明したステップS1004～ステップS1008を繰り返した後の状態は、以下ようになる。

【0073】即ち、1ページ分の描画に必要なデータは、以下の2つの要素から構成されることになる。

1. 描画順序に従ってキューイングされた中間コード
この中間コードに基づいて、制御部602は、バンドイメージを高速に描画（レンダリング）できる。

【0074】2. 描画順序に従ってキューイングされた圧縮バンドイメージデータ

ステップS1010では、印字を開始するために、プリンタエンジンの起動をかける。ステップS1011では、プリンタエンジンに起動がかかったならば、制御部602は、印字処理を開始する。この印字処理は、キューイングされている中間コードに基づいて、DRAM604のバンドバッファに対して、イメージ描画（レンダリング）を行う処理である。

【0075】ここでは、セクタ606、メモリコントローラ603経由のアクセスとなり、セクタ606内のL4、L5、L6の各データパスを確立する（図7）。尚、圧縮されたバンドイメージについては、その圧縮データポイント等がキューイングされている。また、非圧縮のバンド描画中間コードについては、上記従来例で述べたようにダブルバッファ構成にて処理を行なう場合、2バンド分のイメージのレンダリングが先行される。そのため、それらのレンダリングが終了してからプリンタエンジンに起動を掛ける方が、オーバランに対しては有利である。

【0076】この状態で印字動作が進み、ステップS1012では、現状の印字位置に対する描画データソースの種類を、キューイングされたデータから判断し、圧縮されているデータソースの印字であれば、ステップS1014へ、そうでなければ、ステップS1013へと進む。ステップS1013では、既に、ステップS1011でイメージデータ生成を終了、もしくは、ステップS1010でのエンジンの起動以前に生成してあるバンドイメージデータをエンジン609側に送出して、1バンドイメージの印字を実現する。ここで、エンジン609側への送出は、セクタ606、DMA607、エンジンインターフェース608を介して行う。この場合のセクタ606のデータパスは、非圧縮／伸長パス（図7）となる。次に、ステップS1015へ進む。

【0077】一方、ステップS1014では、DRAM604の圧縮データ格納エリアのデータをソースデータとし、セクタ606を伸長処理時のパス（図7）に設定し、DMA607（705）とエンジンインターフェース608を用いて、エンジン609のデータ転送レートに合わせて、リアルタイムで伸長処理を施しつつ、1バンドイメージ印字を実現する。

【0078】ステップS1013、もしくは、ステップS1014によって（1つのバンドイメージ印字処理は何れか一方の処理となる）1バンドイメージの印字が終了されたならば、1ページ分のバンドイメージの印字が

終了したか否かを判断し（ステップS1015）終了していなければ、ステップS1011のステップへ戻り、1ページ終了であれば1ページ分の印字処理の終了となる。

【0079】ステップS1011のステップに戻った時点で印字終了したバンドイメージ印字処理が伸長処理による場合（ステップS1014）は、ステップS1011のステップはスルーする。また、非伸長処理8ステップS1013）の場合は、既に前のステップS1013にてバンドイメージの印字を終了しているエリアに対して、キューイングされた処理が中間コードからのラスターライズを含む場合、制御部602が、次のバンドイメージ生成（ラスターライズ処理）をバックグラウンドにて行なう。

【0080】一方、バックグラウンドによるラスターライズの最中には、次のバンド印字のためにダブルバッファにて既に生成されているイメージデータを、ステップS1013を通じて印字するか、ステップS1014のステップでリアルタイムの伸長処理印字を行なっているかのいずれかの状態になるのも上記実施の形態と同様である。

【0081】以上のような制御の流れ、オーバランに対する処理判断を行なっても機器全体のスループットを向上させつつ、ほぼ上記実施の形態と同様の効果が期待できる。更に他の実施の形態として挙げた中のオーバラン判断の比較になるあらかじめ設定された中間コードデータ量を示す数値については、ROMの形で後から変更不能とすることなく、ユーザが任意に設定したり、オーバランエラーの頻度を本プリンタコントローラが学習し変化させていくことも考えられ、やはり同様の効果が期待できる。

【0082】一方更にスループットを向上させる目的で、ジョブ単位にユーザがもしくは一般のデータ供給源であるホストコンピュータ上で動作するドライバソフトウェア、ユーザが操作するアプリケーションソフトウェア等により、プリンタコントローラに対し非圧縮／伸長処理（バンド処理）で印字を行うか、圧縮／伸長処理にて作業を行うか指定できるようにしてもよい。

【0083】実際に印字を行なうユーザはその印字内容が複雑であるか否かを知っているわけであり、例えば簡単なドキュメントをプリンタコントローラ内に備わるフォントを用いて印字するのであればすべてを非圧縮／伸長処理（バンド処理）にて行ない、その分プリンタ自体のスループットを上げ素早く印字した方が良好と言えるケースがあり、ここに示した対応はよりフレキシビリティに富んだ機器の供給に貢献できることになる。

【0084】又、圧縮／伸長処理か、非圧縮／伸長処理かの判断については、これまでに述べてきた印字すべきデータの内容により処理を切替えることに加え、機器に搭載された実装メモリ容量の変化（ユーザがメモリ自体

を拡張したケースや、ダウンロードフォントの容量等の要素変化）を考慮し、その内容により処理を変化させれば更にフレキシビリティに富んだ機器の構築が可能である。

【0085】例えばユーザがメモリを拡張した、もしくはユーザによりダウンロードされるデータ量が少ない、従って使用できるメモリ容量が多い場合、本実施の形態での非圧縮／伸長処理では、特に限定するものではないが2バンドバッファ構成のプリンタコントローラ及び描画処理として記してはきたが、これを3バンド、4バンドと言うようにバンドバッファ数を増やし、その分プリンタエンジン起動前にあらかじめイメージ描画（レンダリング）しておけるバンド数を増やし、結果として圧縮／伸長処理の頻度を下げオーバランを回避しつつスループットを上げた印字の可能な環境を提供できることになる。

【0086】又更にユーザ拡張等により、制御部がイメージ生成（レンダリング）する際に競合バスマスタを付加したならば、それらによるオーバヘッドを考慮し、イメージ生成（レンダリング）時間をより正確に計算する等の拡張も考えられる。なお、本発明は、複数の機器（例えばホストコンピュータ、インタフェイス機器、リーダー、プリンタなど）から構成されるシステムに適用しても、一つの機器からなる装置（例えば、複写機、ファクシミリ装置など）に適用してもよい。

【0087】また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを

読み出し実行することによっても、達成されることは言うまでもない。

【0088】この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。プログラムコードを供給するための記憶媒体としては、例えば、フロッピーディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

【0089】また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0090】さらに、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わる

メモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0091】本発明を上記記憶媒体に適用する場合、その記憶媒体には、先に説明したフローチャートに対応するプログラムコードを格納することになるが、簡単に説明すると、図11のメモリマップ例に示す各モジュールを記憶媒体に格納することになる。すなわち、少なくとも、画像を入力してメモリに格納するステップS801ーS802の処理に対応する「画像を入力モジュール」、入力データからバンド処理可能な中間データを生成するステップS803の処理に対応する「中間データ生成モジュール」、中間データに基づいて、画像形成時のオーバーランを推定するステップS804の処理に対応する「オーバーラン推定モジュール」、オーバーランが推定されたとき、中間データをビットマップデータに変換するステップS806の処理に対応する「中間データービットマップデータに変換モジュール」、ビットマップデータを圧縮するステップS807の処理に対応する「圧縮モジュール」、1頁分の中間データ、あるいは、圧縮データが生成されたらプリンタエンジンの起動をかけるステップS809の処理に対応する「プリンタエンジン起動モジュール」、圧縮データに基づいて画像形成する場合は、その圧縮データを伸張する「伸張モジュール」、中間データに基づいて画像形成する場合は、中間データに対応する描画データを生成してプリンタエンジンに出力し、圧縮データに基づいて画像形成する場合は、伸張されたデータをプリンタエンジンに出力する「描画データ出力モジュール」の各モジュールのプログラムコードを記憶媒体に格納すればよい。

【0092】

【発明の効果】以上説明したように本発明によれば、オーバーランが生ずることなく、高品位な画像形成を安価な構成で実現できる。

【図面の簡単な説明】

【図1】一般的なプリントシステムの概略構成を示すブ

ロック図である。

【図2】バンド処理を行なう従来の構成を示したブロック図である。

【図3】バンド処理を行なう従来の制御の流れを示したフローチャートである。

【図4】圧縮／伸長技術を盛り込んだ従来の構成を示すブロック図である。

【図5】圧縮／伸長技術を盛り込んだ従来の制御の流れを示したフローチャートである。

10 【図6】本発明の1実施の形態として、オーバーラン予測に基づく処理の切り換えを実現したハード構成を示したブロック図である。

【図7】本発明の1実施の形態として、オーバーラン予測に基づく処理の切り換えを実現したハード構成の内、セクタによるデータパスの変化を示した図で、制御部から圧縮部を通じて圧縮データを格納する場合のデータパス構成を示した図である。

20 【図8】本発明の1実施の形態として、オーバーラン予測に基づく処理の切り換えを実現した場合の制御の流れを示すフローチャートである。

【図9】制御部が描画したイメージデータを圧縮部経由でDRAMに格納するのではなく、一旦バンドイメージデータとしてDRAM内のバンドバッファにラスタライズ終了後、1バンド分を一括して圧縮し格納する際のセクタ部のデータパスを示した図である。

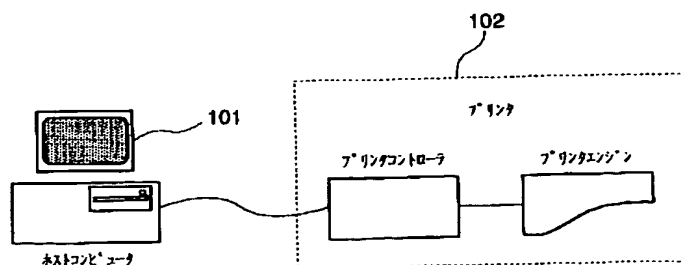
【図10】オーバーランの判断に処理時間の算出ではなく中間コードのデータ量を用いた場合の制御の流れを示したフローチャートである。

30 【図11】コンピュータ読み取り可能な記録媒体に格納された各プログラムモジュールのレイアウトの一例を示す図である。

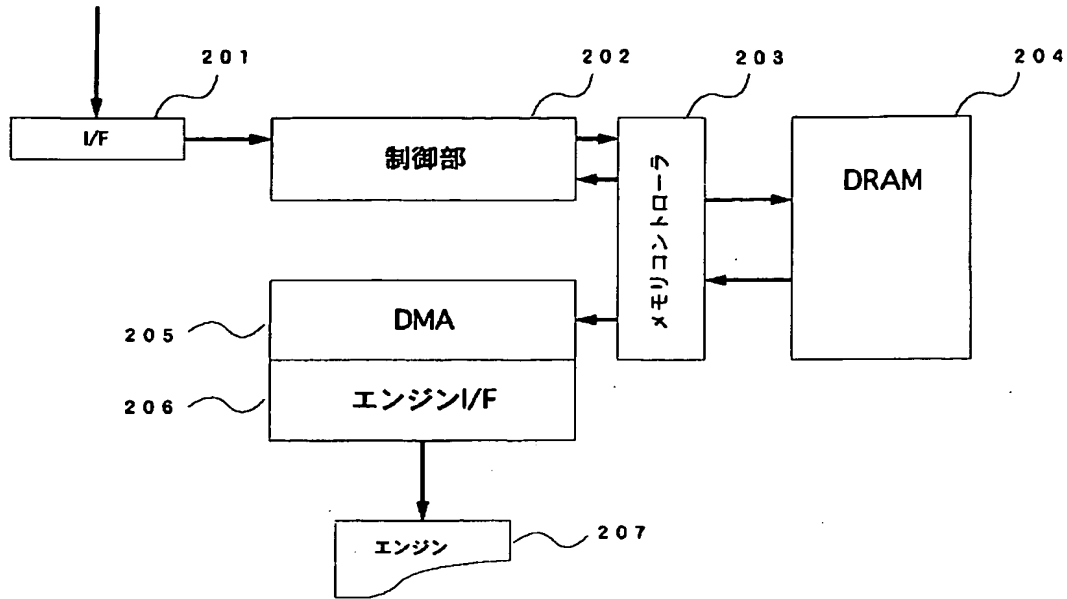
【符号の説明】

701 制御部
702 圧縮／伸張部
703 セクタ
704 メモリコントローラ
705 DMA

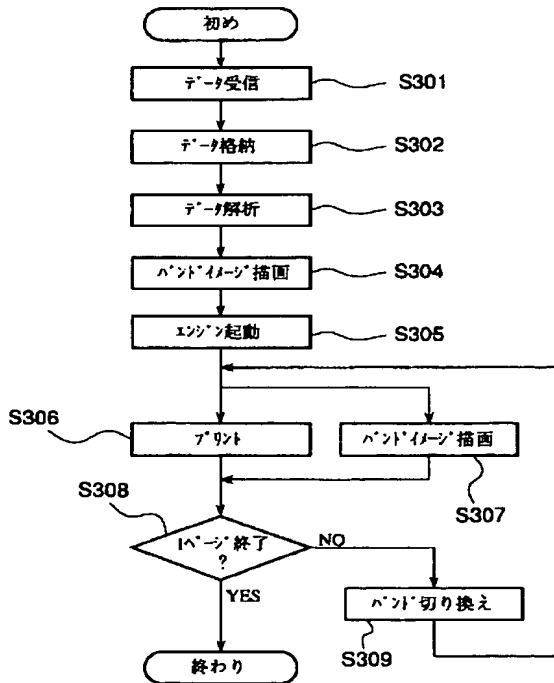
【図1】



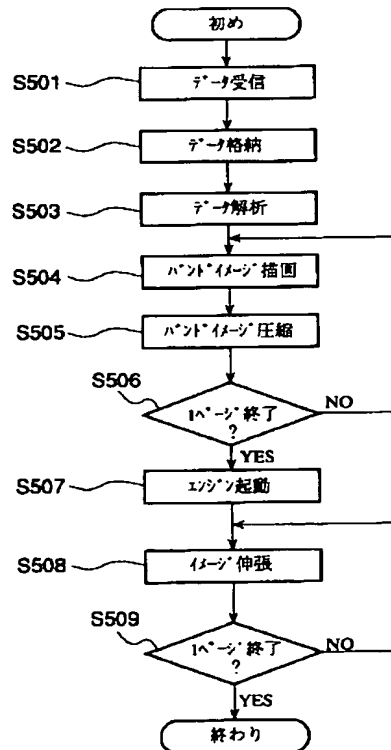
【図2】



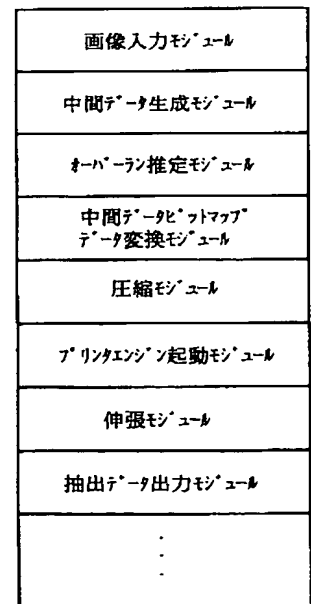
【図3】



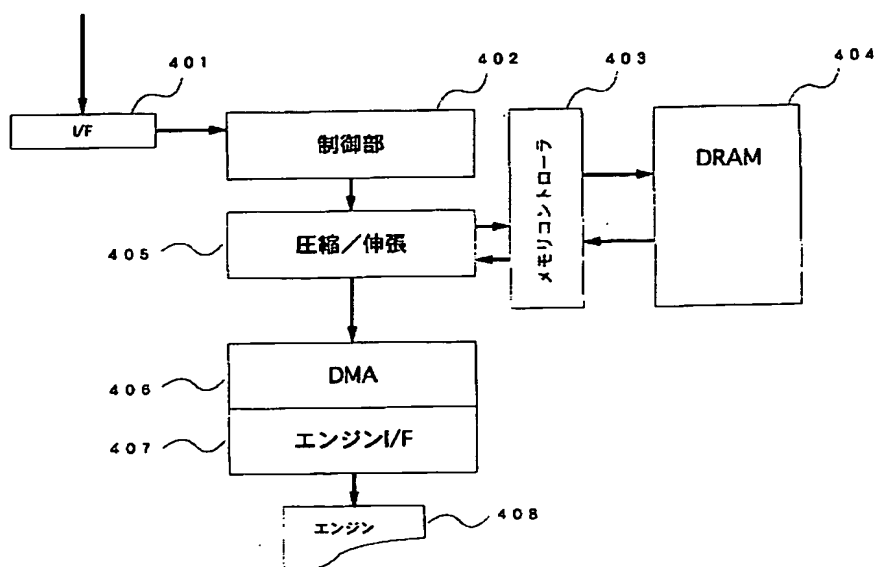
【図5】



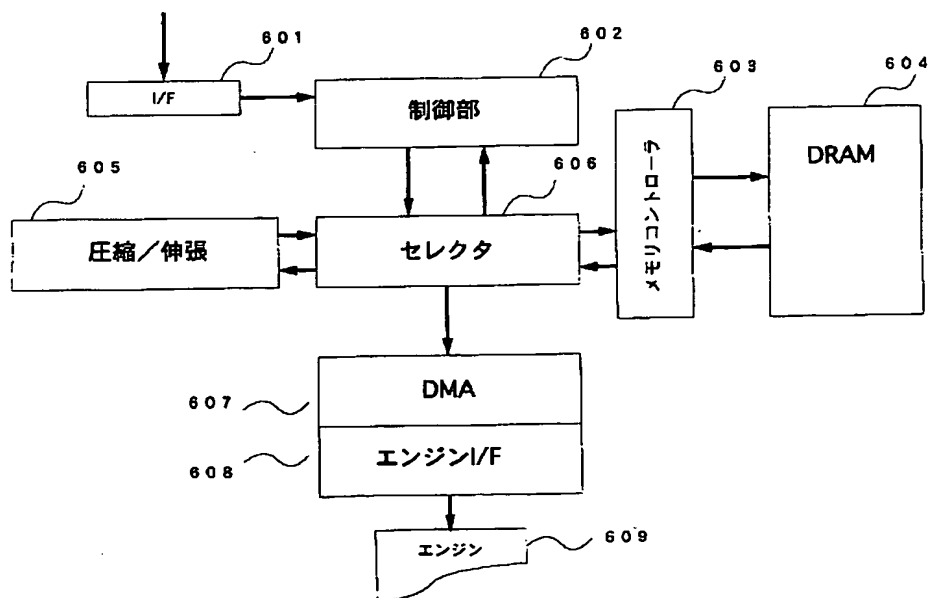
【図11】



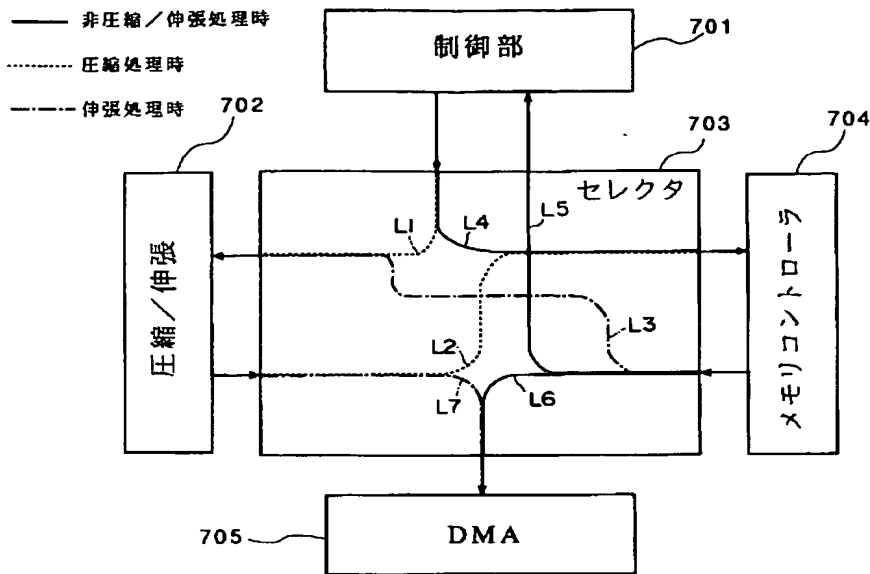
【図4】



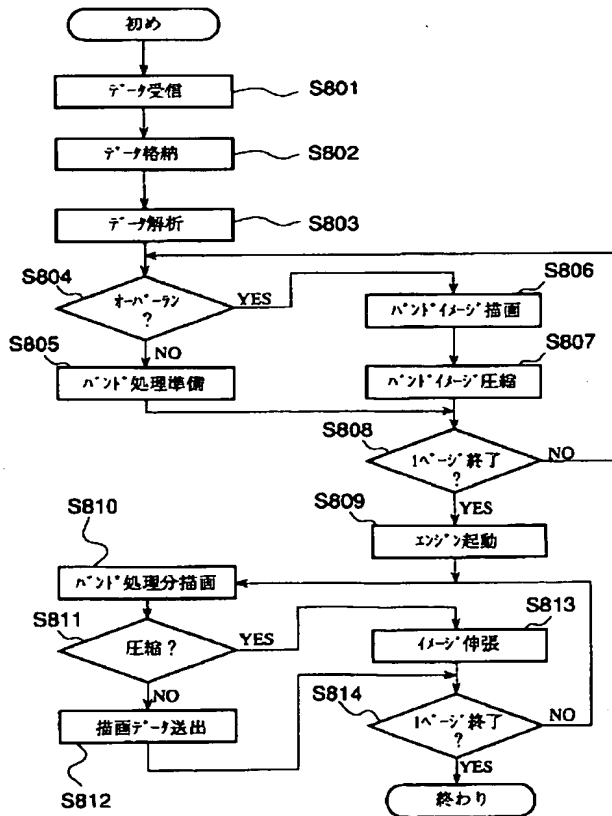
【図6】



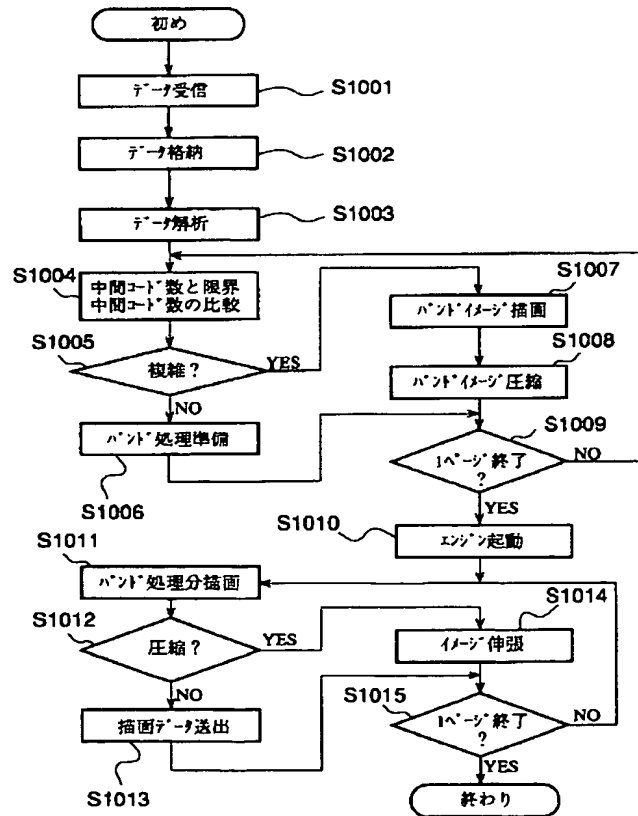
【図7】



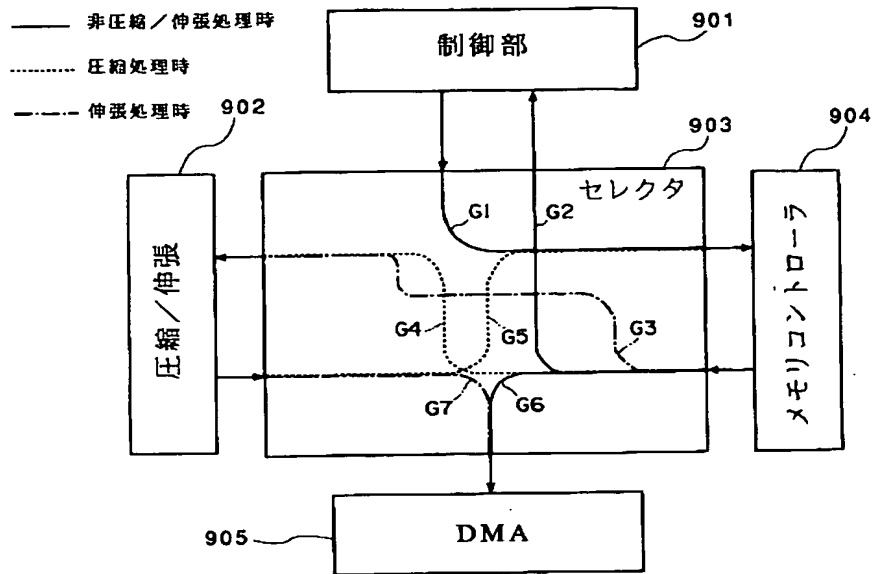
【図8】



【図10】



【図9】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.